templates - a tool to improve productivity

Srini, rs@toprllc.com

July 25, 2020

1 Introduction

There are no shortcuts to gain proficiency in any discipline - particularly programming languages - except to practise a lot. Over the course of a self directed program, you will find yourself adapting a pattern and try to apply the pattern over and over again as you explore different facets. For example, a command line interface with switches and options might be a technique to explore ideas in encryption, compression, text processing and so on. A template of a **cli** program could be a great starting point for each project.

The ubiquitous web development framework **Node.js** `https://nodejs.org/en/about/` for example supports a number of tools for the beginner to setup an initial project based on a template. With very little effort, a new project can be setup to help build your application rapidly.

In this projectlet, we specify a cli template and a web server template. They are designed to impose a standard on such efforts in addition to providing a starting point for new projects.

2 Command Language Template

Expected features of cli applications   The following is a list of features that would be useful for potential users.

- help. this should print out the usage ie all the options and arguments. In most cases invoking the utility without any arguments should provide a help output and exit.

- version reporting - to enable issue reporting and tracking. Wherever feasible, a program can also provide a way to track the software back to a configuration management system.

- verbosity - either a simple boolean or a verbosity level that requests diagnostic information.

- No passwords to be specified on the command line. When there is a need for passwords an alternate way to specify such as environment variable could be a good solution.

Design guidelines

- Separation of the implementation from the invocation by a command line. Perhaps there may be a different way the service is deployed - possibly with a GUI or behind a web server.

- Judicious use of logging will enhance the ease of use.

- A design that can be adapted to a new requirement. Preferably a simple set of commands - in a script should be able to create a new project with the template as the starting point.

- It should be functional. Though minimal in what it supports, the template should be buildable and perform the simple functions listed above.

2.1 Example

The example in the repository below can be built and is functional.

```
$ ./template

        This is a template project for command line utilities

Usage:
  template [command]

Available Commands:
  help        Help about any command
  version     Report the version of the application

Flags:
      --config string    config file (default is $HOME/.cli/cli.yaml)
  -h, --help             help for template
      --verbose          be verbose
  -v, --version          version for template

Use "template [command] --help" for more information about a command.
```

3 Web server template

Essential Features   One example of deriving an app from the cli template is a web server which itself is a template. In this case, the requirements are:

- Provide some set of services using a standard protocol such as **https** implying transport layer security.

- Compatible with web browsers found on standard platforms

- Support the notion of an admin user with support for administrative functions and a normal user with application functions. Before providing any service, the server shall require a login.

- Support a basic user database with usernames and passwords. While the database itself may be a simple file, the passwords should not be stored in cleartext.

- Ability to support development without a production oriented environment. This may call for self signed certificates, operate under different platforms.

- Basic setup for serving static html content.

3.1 Minimal Realization based on the cli template

The repository contains a template of a https server to meet the above. As the example illustrates, it can be **installed** when a self signed certificate is generated to support the **TLS** requirement. The installation process also generates a default user database with a minimal set of usernames. The tool prompts for the default passwords.

3.2 Usage

```
$ ./main --help

        Generic TLS server

Usage:
  server [flags]
  server [command]

Available Commands:
  help        Help about any command
  install     Installs the server
  show        Show details
  user        User admin
  version     Report the version of the application

Flags:
      --config string   config file name. Default is
  -h, --help            help for server
  -v, --verbose int     verbosity level 1 .. 16
      --version         version for server

Use "server [command] --help" for more information about a command.
```

3.3 Installation Example

```
$ ./main install
Server directory[C:\msys64\home\RajaS/server] :
Server Port[443] :
Server URL[localhost] :
Admin Password (username: admin)[admin] :
User Password (username: user)[user] :
2020/07/25 07:21:53 Saved Configuration file C:\msys64\home\RajaS/server/etc/server.yaml
2020/07/25 07:21:53 Generating certificates
2020/07/25 07:21:55 Created private key file C:\msys64\home\RajaS/server/etc/keypair.pvt.pem
2020/07/25 07:21:55 Created public key C:\msys64\home\RajaS/server/etc/keypair.pub.pem
2020/07/25 07:21:55 Creating Cert
2020/07/25 07:21:55 Loaded C:\msys64\home\RajaS/server/etc/keypair.pvt.pem and created a private key
2020/07/25 07:21:55 Certificate saved to C:\msys64\home\RajaS/server/etc/certfile
2020/07/25 07:21:55 Setting up website
2020/07/25 07:21:55 Installing files of templates to C:\msys64\home\RajaS/server/html
2020/07/25 07:21:55 Installed footer.html in C:\msys64\home\RajaS/server/html
2020/07/25 07:21:55 Installed header.html in C:\msys64\home\RajaS/server/html
2020/07/25 07:21:55 Installed index.html in C:\msys64\home\RajaS/server/html
2020/07/25 07:21:55 Installed login.html in C:\msys64\home\RajaS/server/html
2020/07/25 07:21:55 Installed menu.html in C:\msys64\home\RajaS/server/html
2020/07/25 07:21:55 Installed stats.html in C:\msys64\home\RajaS/server/html
```

4 Implementation

The following repository contains 2 projects - a cli template and a webserver template in the **go** language.

Implementation in **go** : `https://gitlab.com/projtemplates/go.git`